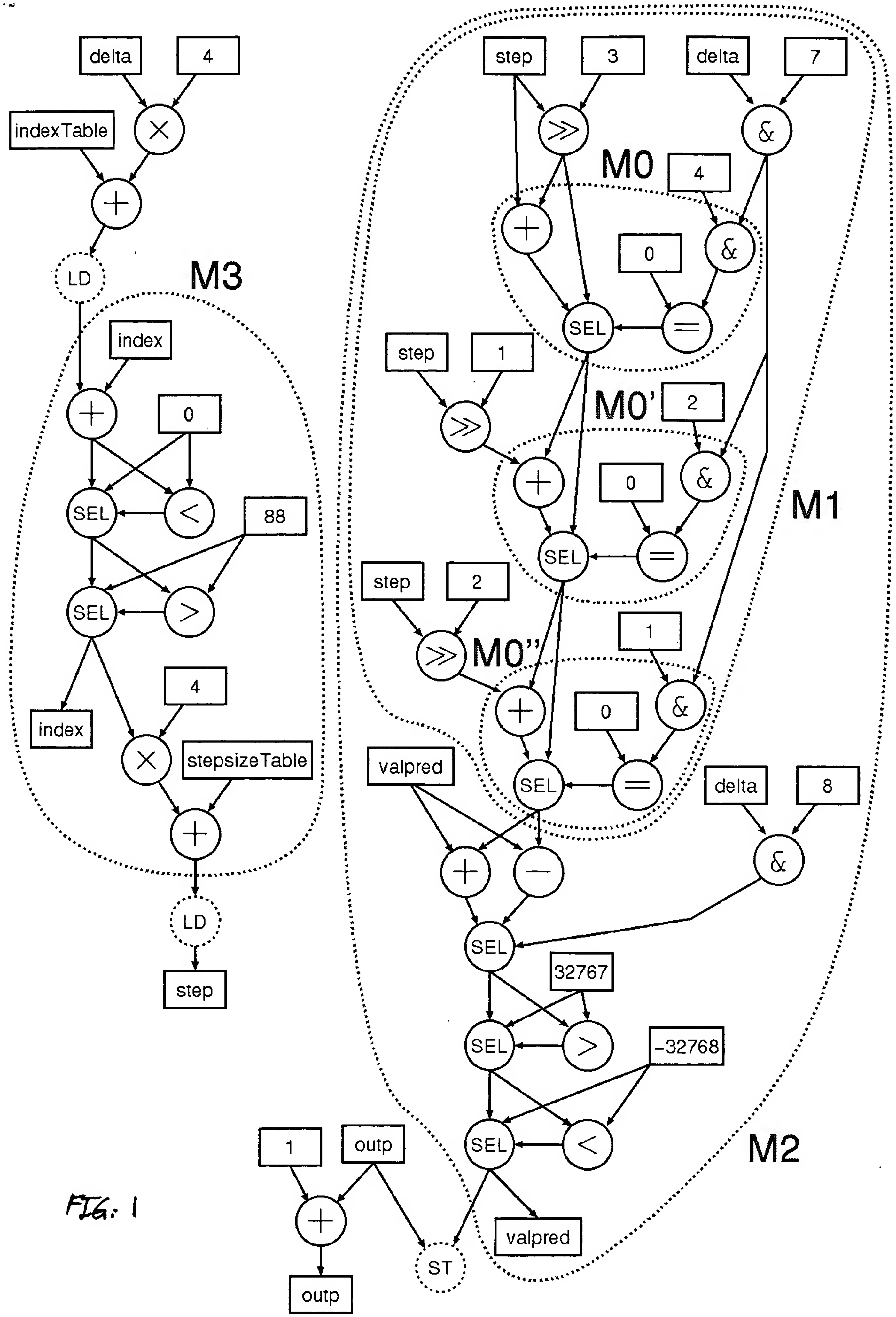
delta
4
indexTable
LD
M3
index
0
88
index
4
stepsizeTable
LD
step
step
3
delta
7
M0
4
0
SEL
step
1
M0'
2
0
SEL
M1
step
2
M0''
1
0
SEL
valpred
delta
8
SEL
32767
SEL
-32768
SEL
M2
1
outp
valpred
ST
outp

FIG: 1

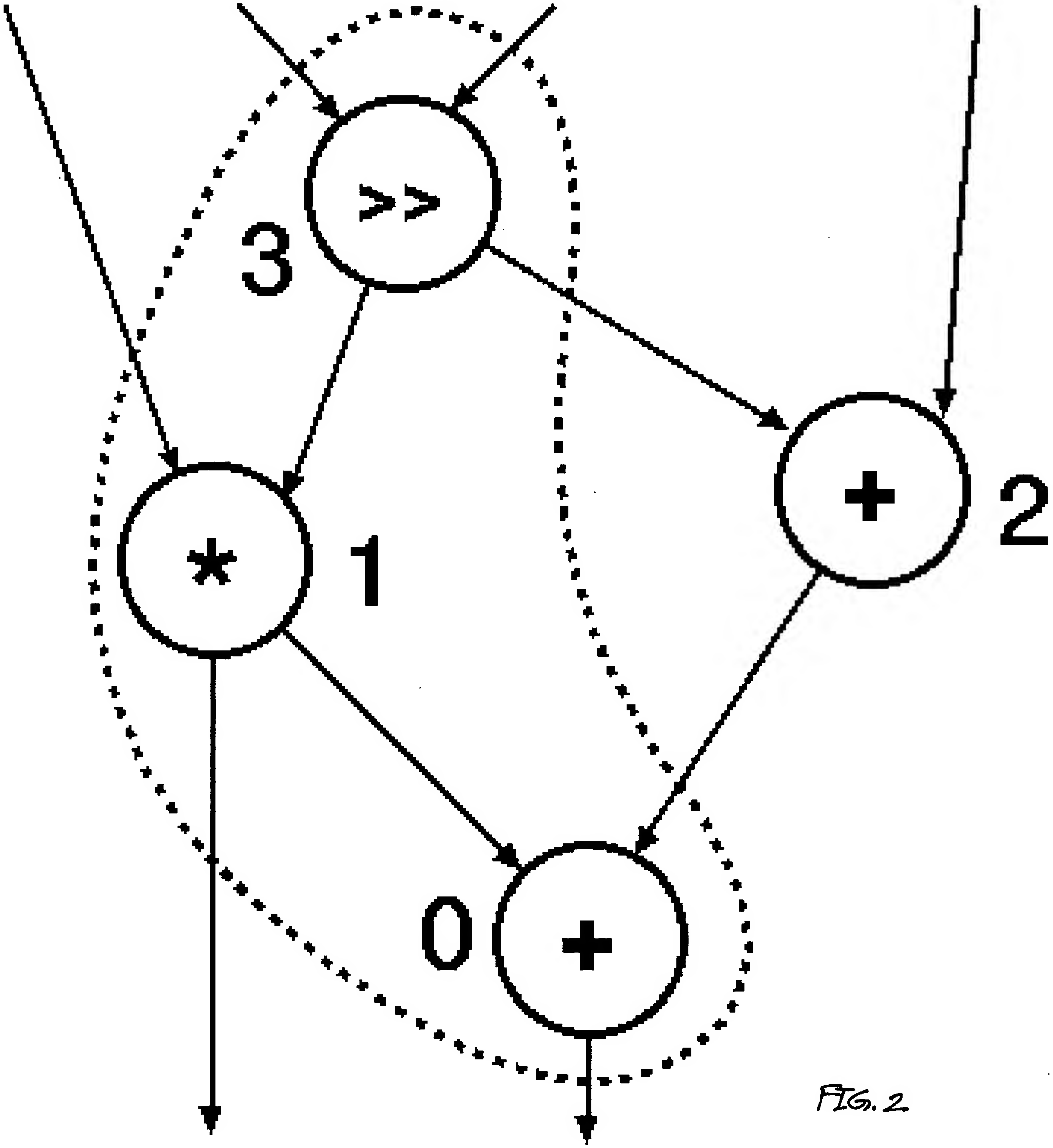

FIG. 2

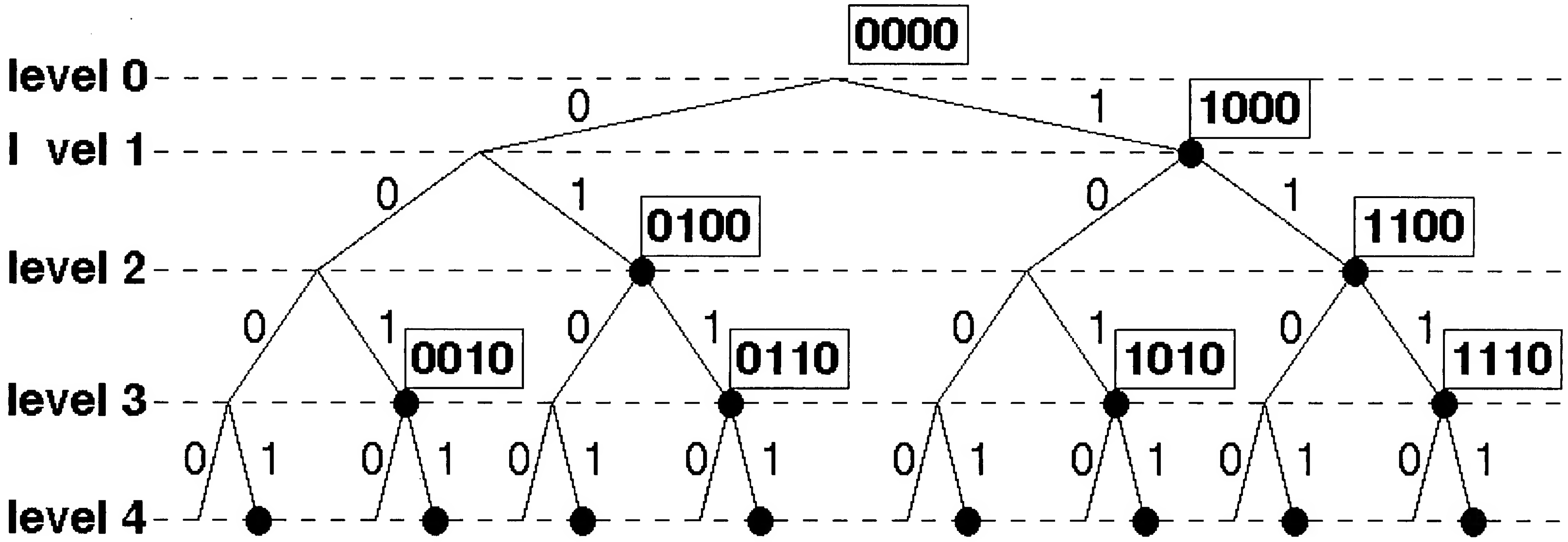
0000
level 0
0
1
1000
l vel 1
0
1
0
1
0100
1100
level 2
0
1
0010
0
1
0110
0
1
1010
0
1
1110
level 3
0
1
0
1
0
1
0
1
0
1
0
1
0
1
0
1
level 4

FIG. 3

```
identification() {
   for (i = 0; i < NODES; i++) cut[i] = 0;
   topological_sort();
   search(1, 0);
   search(0, 0); }

search(current_choice, current_index) {
   cut[current_index] = current_choice;
   if (current_choice == 1) {
      if (!output_port_check()) return;
      if (!convexity_check()) return;
      if (input_port_check()) {
         calculate_speedup();
         update_best_solution(); } }
   if ((current_index + 1) == NODES) return;
   current_index = current_index + 1;
   search(1, current_index);
   search(0, current_index); }
```

Figure 4: The identification algorithm.

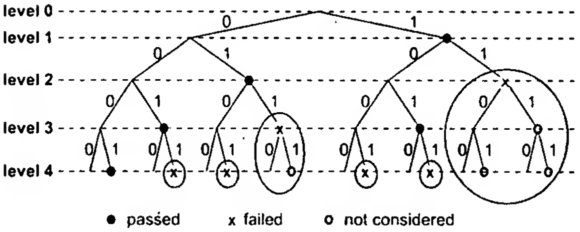

Figure 5: The execution trace of the algorithm for the graph given in Figure 2 and $N_{out} = 1$.

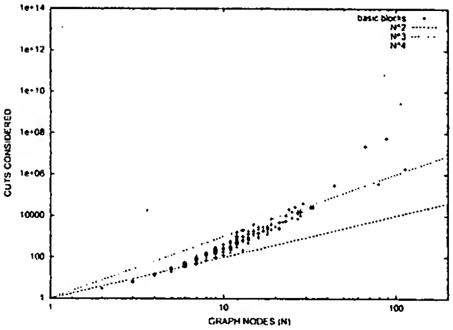

Figure 6: Number of cuts considered by the algorithm with $N_{out} = 2$ and any N_{in}.

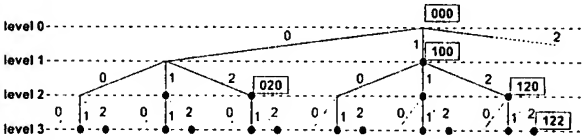

Figure 7: A search tree for two cuts.

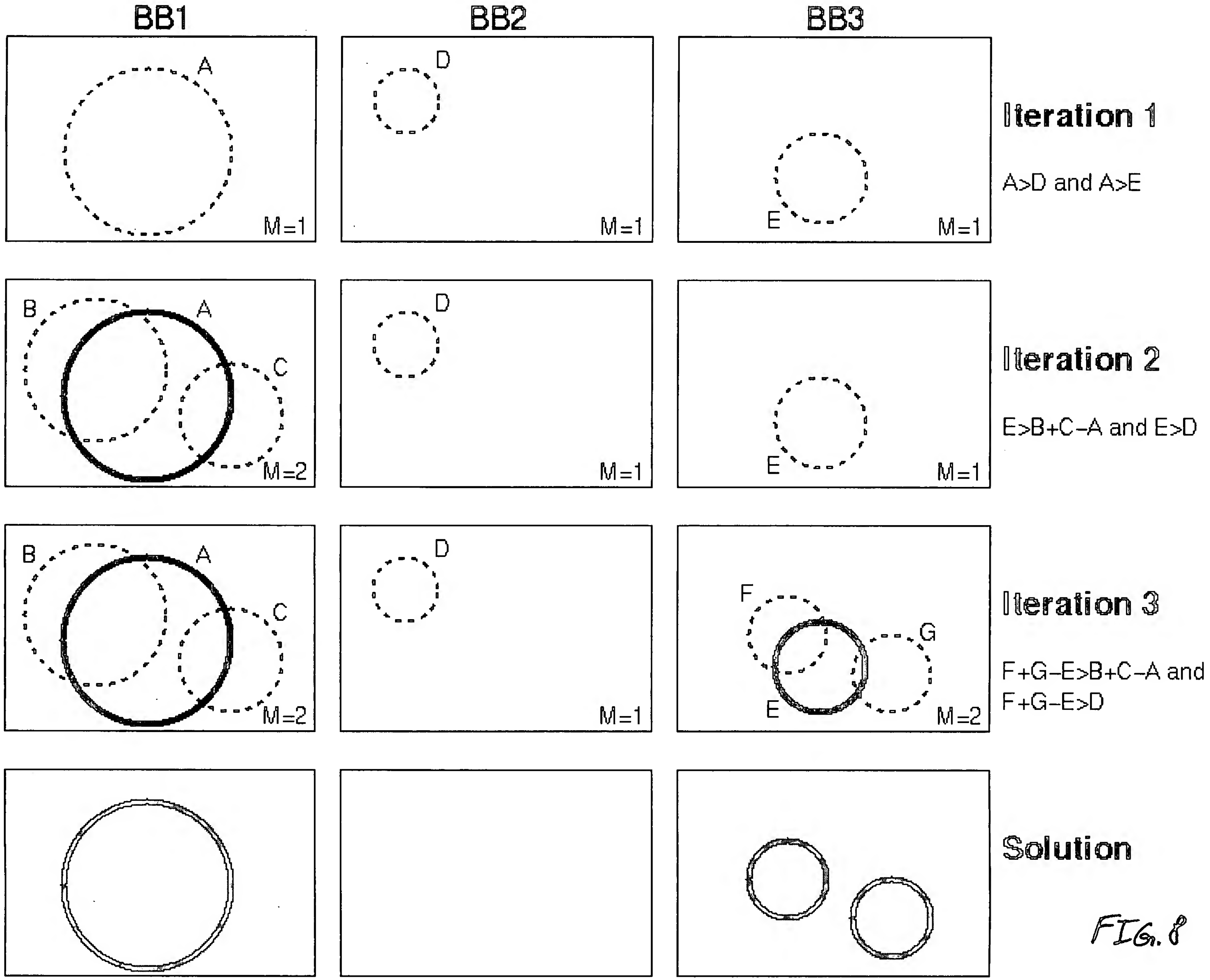
BB1
BB2
BB3
A
D
E
M=1
M=1
M=1
Iteration 1
A>D and A>E
B
A
C
M=2
D
M=1
E
M=1
Iteration 2
E>B+C–A and E>D
B
A
C
M=2
D
M=1
F
G
E
M=2
Iteration 3
F+G–E>B+C–A and
F+G–E>D
Solution

FIG. 8